



# KOREAN PATENT ABSTRACTS(KR)

Document Code: A

(11) Publication No. 1020000039210

(43) Publication Date. 20000705

(21) Application No. 1019980054475

(22) Application Date. 19981211

(51) IPC Code:

H04B 1/69

(71) Applicant:

SAMSUNG ELECTRONICS CO., LTD.

(72) Inventor:

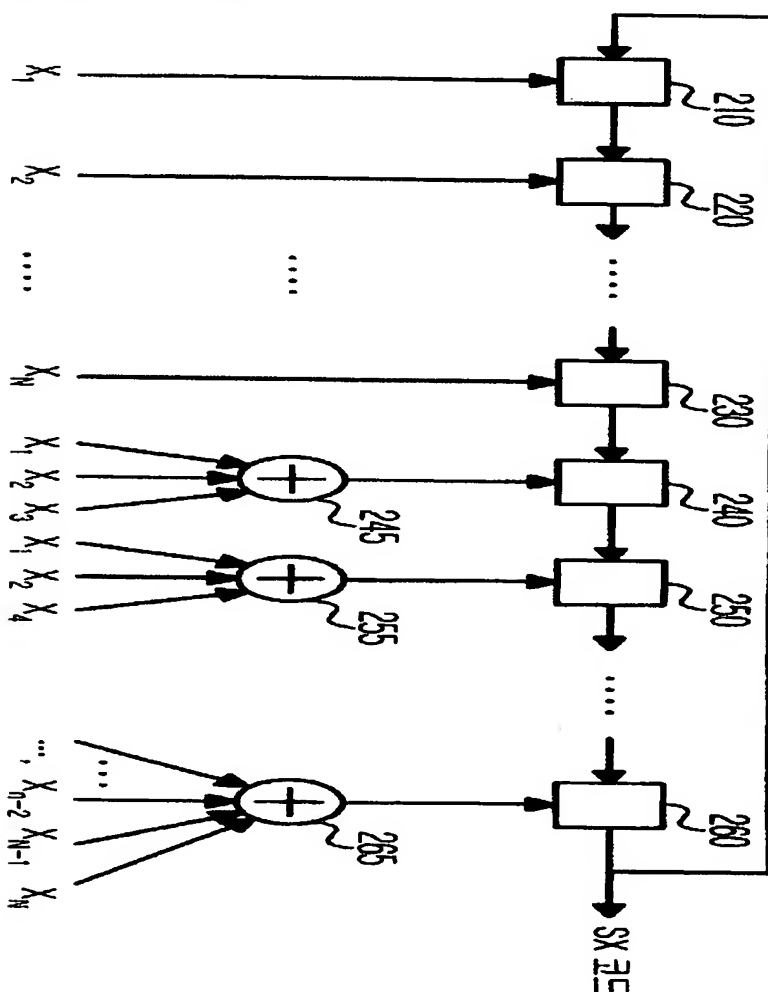
KANG, DONG HUI

(30) Priority:

(54) Title of Invention

METHOD FOR GENERATING DIFFUSION CODE HAVING ORTHOGONAL CHARACTERISTIC

Representative drawing



(57) Abstract:

PURPOSE: A method for generating diffusion code having orthogonal characteristic is provided to reduce timing required for generating code, generate an orthogonal characteristic by using Ex-OR logic without using additional control logic, reduce the time required for generating code in the system by applying SX code to a code division multiple system, and simplify the structure of the system, thereby improving the efficiency.

CONSTITUTION: Input symbols are received. A subcode performs exclusive OR to an odd number of bit selected among the input symbol. A SX code is generated by using possible all odd number of Ex-OR subcode. Input data is diffused using the SX code.

COPYRIGHT 2000 KIPO

if display of image is failed, press (F5)



## 발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 확산 스펙트럼 통신(Spread Spectrum Communication)을 위한 확산코드(Spread Code) 발생방법에 관한 것으로서, 특히 코드분할 다중접속(Code Division Multiple Access: CDMA) 기술을 사용하는 이동 전화기(Mobile Telephone) 또는 기지국(Base Station)에서 사용자 채널의 구분 및 코드의 직교특성(Orthogonality)을 유지하기 위한, 직교특성을 가지는 확산코드 발생방법에 관한 것이다.

코드분할 다중접속 시스템은, 공통된 주파수를 사용하는 각 가입자들의 신호를 송신하기 전에 각자의 고유한 확산코드(Spread Code)를 곱하고 대역을 확산시켜 하나의 주파수에 실어 보낸다. 수신할 때는 먼저 역확산을 수행하고, 송신에서와 똑같은 코드를 곱하여 각각의 신호를 구분한다. 코드분할 다중접속 시스템은 확산 대역을 사용함으로써, 주파수의 사용 효율을 높이고, 코드를 사용하여 암호화를 지원한다.

코드분할 다중접속 시스템에서 기지국(Base Station)에서 단말국(Radio Terminal)으로 향하는 채널을 순방향 링크(Forward Link)라 하며, 단말국에서 기지국으로 향하는 채널을 역방향 링크(Reverse Link)라 한다. 순방향 링크에서 기지국은 이동 교환국으로부터 전해진 신호에 직교특성을 가지는 의사잡음 코드(Pseudorandom Noise Code: PN Code)를 곱하여 확산시킴으로써, 채널을 구분한다. 또한 역방향 링크에서 이동 전화기는 다수의 입력데이터간 직교특성을 유지하기 위하여 확산코드를 사용한다.

일반적으로 코드분할 다중접속 기술을 사용하는 이동 전화기나 기지국은 직교특성을 가지는 확산코드로서 왈시코드(Walsh Code)나 하다마드 코드(Hadamard Code)를 사용한다. 입력신호에 따른 직교특성을 유지하면서 왈시코드나 하다마드를 빠른 시간 내에 생성하기 위해서, 종래 기술에 의한 확산코드 발생방법은, 빠른 하다마드 코드기법(Fast Hadamard Coding)을 사용한다.

도 1은 종래 기술에 의한 하다마드 코드 발생방법의 일 실시예를 나타낸 것으로, 도시한 바와 같이, 클럭에 의하여 4비트 출력(A0 ~ A3)을 생성하는 카운터(110)와; 2비트 입력신호를 받아 일정한 규칙에 의해 4비트 출력(B0 ~ B3)을 생성하는 버퍼(120); 상기 버퍼의 출력을 제어하는 제어로직(130); 상기 카운터의 출력신호(A0 ~ A3)와 버퍼의 출력신호(B0 ~ B3)를 각각 논리곱(And)하는 4개의 앤드 게이트(140)(142)(144)(146) 및 상기 4개의 앤드 게이트의 출력을 배타적 논리합(Exclusive-OR)하여 출력을 생성하는 Ex-OR 게이트(150)로 구성된다.

상기 버퍼(120)는 제어로직에 의하여 4비트 출력신호(B0 ~ B3)를 선택적으로 출력하는데, 2비트 입력에 대한 4비트 출력의 예는 표 1과 같다.

[표 1]

입력		출력			
		B0	B1	B2	B3
0	0	0	0	0	0
0	1	0	1	0	1
1	0	0	0	1	1
1	1	0	1	1	0

코드분할 다중접속 시스템의 기지국이나 이동 전화기는, 채널마다 미리 할당된 행(row)을 곱함으로써, 각 채널을 구분한다. 예를 들어 파일럿 채널은 첫 번째 행을 할당받는다.

상기된 바와 같이 동작하는 종래 기술에 의한 확산코드 발생방법에 있어서는, 입력에 대하여 생성되는 직교코드를 제어하기 위한 제어로직과 버퍼가 필요하기 때문에 코드발생에 시간이 많이 걸리며, 입력에 대하여 생성되는 행을 저장할 메모리 자원이 필요하기 때문에 회로의 구성이 복잡하다는 문제점이 있었다.

## 발명이 이루고자 하는 기술적 과제

따라서 상기한 바와 같이 동작되는 종래 기술의 문제점을 해결하기 위하여 창안된 본 발명의 목적은, 부가적인 제어로직을 사용하지 않고 순수하게 Ex-OR 로직만을 사용하여 직교특성을 가지는 서브조합 Ex-OR(Subset Ex-OR: SX) 코드를 생성하는, 직교특성을 가지는 확산코드 발생방법을 제공하는 것이다.

본 발명의 다른 목적과 장점은 하기된 발명의 상세한 설명을 읽고 첨부된 도면을 참조하면 보다 명백해질 것이다.

## 발명의 구성 및 작용

상기한 바와 같은 목적을 달성하기 위하여 창안된 본 발명에 따른 직교특성을 가지는 확산코드 발생방법의 바람직한 실시예는,

입력신호를 수신하는 과정과;

입력신호 가운데 선택된 홀수개의 비트를 배타적 논리합(Ex-OR)한 서브코드를 생성하는 과정; 및

가능한 모든 홀수개의 Ex-OR 서브코드를 사용하여 SX 코드를 생성하는 과정;을 포함한다.

본 발명에 따른 직교특성을 가지는 확산코드 발생방법의 바람직한 다른 실시예는,

입력신호를 수신하는 과정과;

입력신호 가운데 선택된 홀수개의 비트를 배타적 논리합(Ex-OR)한 서브코드를 생성하는 과정;

가능한 모든 홀수개의 Ex-OR 서브코드를 사용하여 SX 코드를 생성하는 과정; 및

SX 코드를 사용하여 입력 데이터를 확산시키는 과정;을 포함한다.

하기에서 본 발명을 설명함에 있어, 관련된 공지 기능 또는 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략할 것이다. 그리고 후술되는 용어들은 본 발명에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다. 이하 첨부된 도면을 참조하여 본 발명의 바람직한 실시예에 대한 동작 원리를 상세히 설명한다.

본 발명은 통신 시스템에서 데이터를 확산시키기 위하여 사용하는 확산코드에 있어서, 직교특성을 가지는 확산코드를 생성한다. 본 발명에 의해 직교특성을 가지는 확산코드를 SX 코드라고 하면, 스펜(Span) N을 가지는 SX 직교코드는 수학적 식 1과 같이 구현된다.

수학적 식 1

$$SX(X) = (X), F(X), \dots, F(X), \dots, F(X)$$

여기서 스펜 N은 입력가능한 이진 도메인(binary domain)에서의 윈도우 크기(Window size)라고 할 때, 는 윈도우 내에 포함되는 N비트 2진수 형태의 입력신호이며, (\*) 는 i번째 Ex-OR 서브코드(Subcode)이고, 는 Ex-OR 서브코드의 개수로서 이다.

도 2 는 본 발명에 따른 SX 직교코드 생성방법의 일 실시예를 나타낸 것으로서, 도시한 바와 같이, SX 직교코드는 가능한 모든 홀수개의 Ex-OR 서브코드( )를 출력하는 Ex-OR 로직(245)(255)(265)과, Ex-OR 서브코드를 임시저장하는 기억소자(210)(220)(230)(240)(250)(260)에 의하여 생성된다. 각 기억소자는 P개이며, i번째 기억소자는 (\*) 를 저장한다.

일 때, SX 코드에 의해서 생성되는 직교코드는 수학적 식 2에 나타낸 바와 같이 매트릭스(matrix)로 생성된다.

수학적 식 2

$$S = (X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)(X)$$

수학적 식 2에 나타낸 바와 같이, j번째 SX 코드는 매트릭스의 j번째 행(row)이다. 확산코드를 사용하여 데이터를 확산시키는 통신 시스템은 상기 매트릭스의 하나의 행(row)을 하나의 채널에 할당한다. 그러므로 상기 통신 시스템은 최대 P개의 채널을 구분할 수 있다.

이하 스펜 N=4이고, P=8인 직교코드를 만드는 4차의 SX 코드에 대하여 설명하면 다음과 같다. N=4이고 P=8인 직교코드는 수학적 식 3에 의하여 생성되며, 도 3 에 나타낸 바와 같이 구현된다.

수학적 식 3

$$SX(X) = (X), F(X), F(X), F(X), F(X), F(X), F(X), F(X)$$

$$= .X.X.X.X.X.X.X$$

수학적 식 3 및 도 2 에 의해 생성된 8×8 매트릭스 형태의 SX 코드는 도 4 에 나타낸 바와 같다.

이하 스펜 N=4인 SX 코드의 직교특성에 대하여 설명한다. SX 코드의 값들이 수학적 식 4를 만족하면, 그 코드는 직교특성을 가진다. 여기서 기호 는 두 입력이 동일하면 1을 출력하고, 다르면 -1을 출력한다.

수학적 식 4

$$SX = (X) (X) = 0, \text{ 단 } j \neq k$$

, 단

여기서 는 10진수 j를 N비트의 2진수 형태로 나타낸 것이다. 예를 들어, i와 j의 값이 각각 5, 6이면 =(0,1,0,1), =(0,1,1,0)이므로, 와 에 대하여 수학적 식 4를 계산하면 수학적 식 5와 같다.

수학적 식 5

$$SX = (0)$$

즉, SX 코드에 의해 만들어지는 일반적인 코드값(코드행)들은 서로간에 직교특성을 유지한다. 또한 도 3 으로부터 ..... 은 각각 ..... 과 반전관계에 있음을 알 수 있다.

도 5 는 2비트 입력을 사용하여 SX 직교코드에 의해 생성된 2×2 매트릭스이며, 도 6 은 3비트 입력을 사용하여 SX 직교코드에 의해 생성된 4×4 매트릭스이고, 도 7 은 4비트 입력을 사용하여 SX 직교코드에 의해 생성된 8×8 매트릭스이고, 도 8 은 5비트 입력을 사용하여 SX 직교코드에 의해 생성된 16×16 매트릭스이다. 도 5 내지 도 8 에 도시한 바와 같이, SX 직교코드는 2배수로 표현되는 모든 직교특성을 만족하는 매트릭스를 생성한다.

즉, SX 직교코드는 입력 비트들 가운데 선택된 홀수개의 비트를 Ex-OR한 서브코드로 구성된다.

그러므로 상기와 같은 SX 직교코드를 코드분할 다중접속 시스템에 적용함으로써, Ex-OR 로직만을 사용하는 단순한 구조를 사용하여 직교특성을 가지는 확산코드를 발생시킬 수 있다.

본 발명은 다양하게 변형될 수 있고 여러 가지 형태를 취할 수 있으며 상기 발명의 상세한 설명에서는 그에 따른 특별한 실시예에 대해서만 기술하였다. 하지만 본 발명은 상기 발명의 상세한 설명에서 언급된 특별한 형태로 한정되는 것이 아닌 것으로 이해되어야 하며, 오히려 첨부된 청구 범위에 의해 정의되는 본 발명의 정신과 범위 내에 있는 모든 변형물과 균등물 및 대체물을 포함하는 것으로 이해되어야 한다.

발명의 효과

이상에서 상세히 설명한 바와 같이 동작하는 본 발명에 있어서, 개시되는 발명중 대표적인 것에 의하여 얻어지는 효과를 간단히 설명하면 다음과 같다.

본 발명은, 추가적인 제어로직을 사용하지 않고 Ex-OR 로직만을 사용하여 왈시코드나 하다마드코드와 같이 직교특성을 가지는 SX 코드를 생성함으로써, 코드 발생에 필요한 타이밍을 감소시키며, 단순한 구조로 직교 코드를 생성할 수 있는 효과가 있다. 또한 SX 코드를 코드분할 다중접속 시스템에 적용함으로써, 시스템에서 코드를 발생시키는데 필요한 시간을 감소시키며, 시스템의 구조를 단순화하여, 시스템의 효율을 높일 수 있다.

#### (57) 청구의 범위

##### 청구항 1.

입력신호를 수신하는 과정과;

입력신호 가운데 선택된 홀수개의 비트를 배타적 논리합(Ex-OR)한 서브코드를 생성하는 과정; 및

가능한 모든 홀수개의 Ex-OR 서브코드를 사용하여 SX 코드를 생성하는 과정;을 포함하는, 직교특성을 가지는 확산코드 발생방법.

##### 청구항 2.

제 1 항에 있어서, 상기 SX 코드를 생성하는 과정은,

N은 입력신호의 비트수이고,

P는 2의 N-1제곱이고,

j는 10진수 j를 2진수로 표현한 j번째 입력신호이고,

(\*) 는 j번째 Ex-OR 서브코드일 때,

입력신호에 의해 생성되는 P×P 매트릭스형태의 SX 코드 S는,

$$S = (X_1 X_2 X_3 \dots X_N X_1 X_2 X_3 \dots X_N X_1 X_2 X_3 \dots X_N X_1 X_2 X_3 \dots X_N)$$

로 표현되는, 직교특성을 가지는 확산코드 발생방법.

##### 청구항 3.

제 2 항에 있어서,

j번째 입력신호는 입력비트 내지 에 의해 표현될 때,

상기 (\*) 는, 입력비트 내지 가운데 선택된 홀수개의 비트를 Ex-OR한 서브코드인, 직교특성을 가지는 확산코드 발생방법.

##### 청구항 4.

제 2 항에 있어서,

기호를 두 입력이 동일하면 1을 출력하고, 다르면 -1을 출력하는 기호라고 정의할 때,

상기 임의의 SX 코드와 는,

이면  $\sum_{j=1}^N (X_j - 1) = 0$  을 만족하고,

이면  $\sum_{j=1}^N (X_j - 1) = P$  를 만족하는, 직교특성을 가지는 확산코드 발생방법.

##### 청구항 5.

입력신호를 수신하는 과정과;

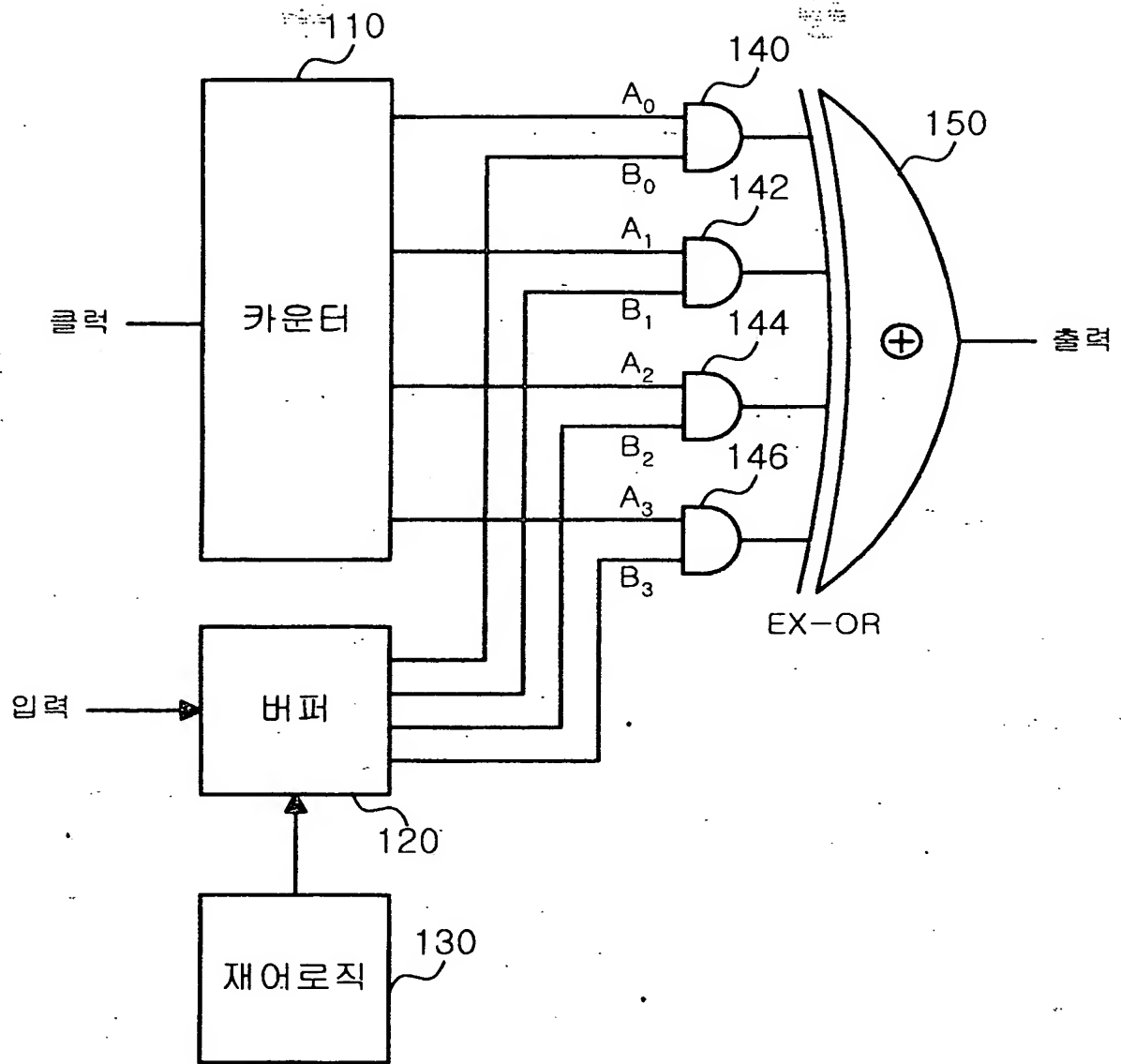
입력신호 가운데 선택된 홀수개의 비트를 배타적 논리합(Ex-OR)한 서브코드를 생성하는 과정;

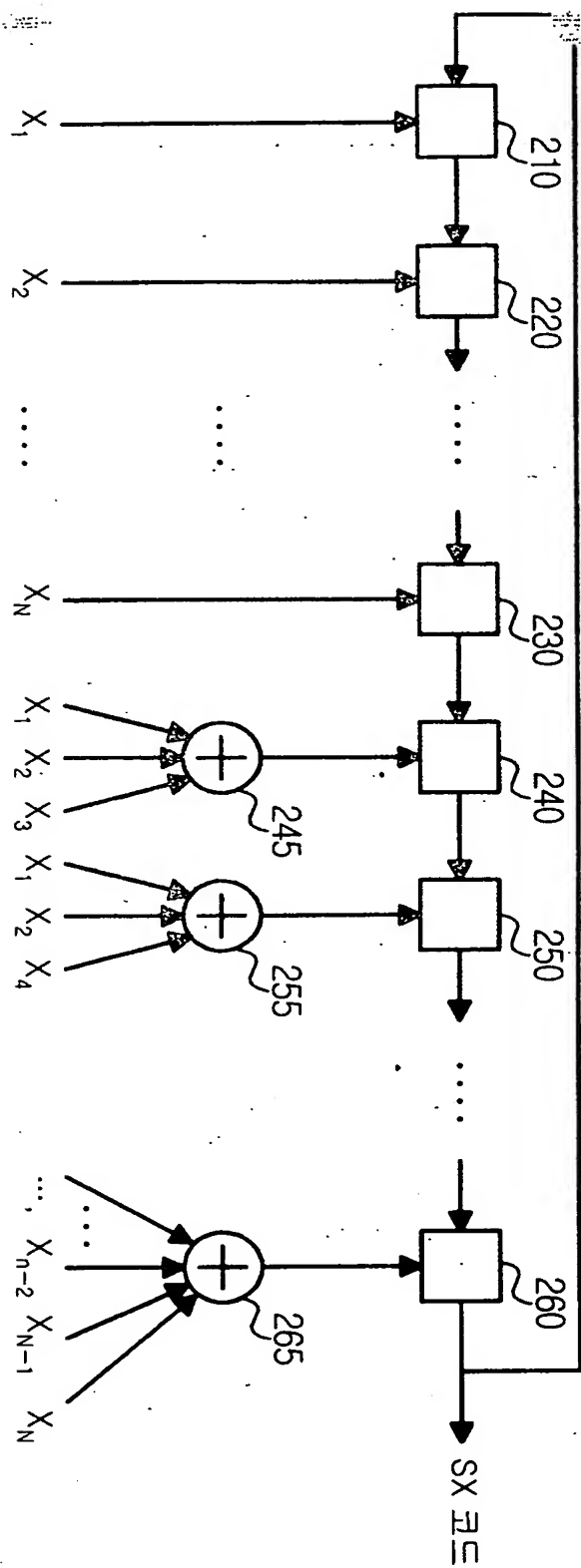
가능한 모든 홀수개의 Ex-OR 서브코드를 사용하여 SX 코드를 생성하는 과정; 및

SX 코드를 사용하여 입력 데이터를 확산시키는 과정;을 포함하는, 직교특성을 가지는 확산코드 발생방법.

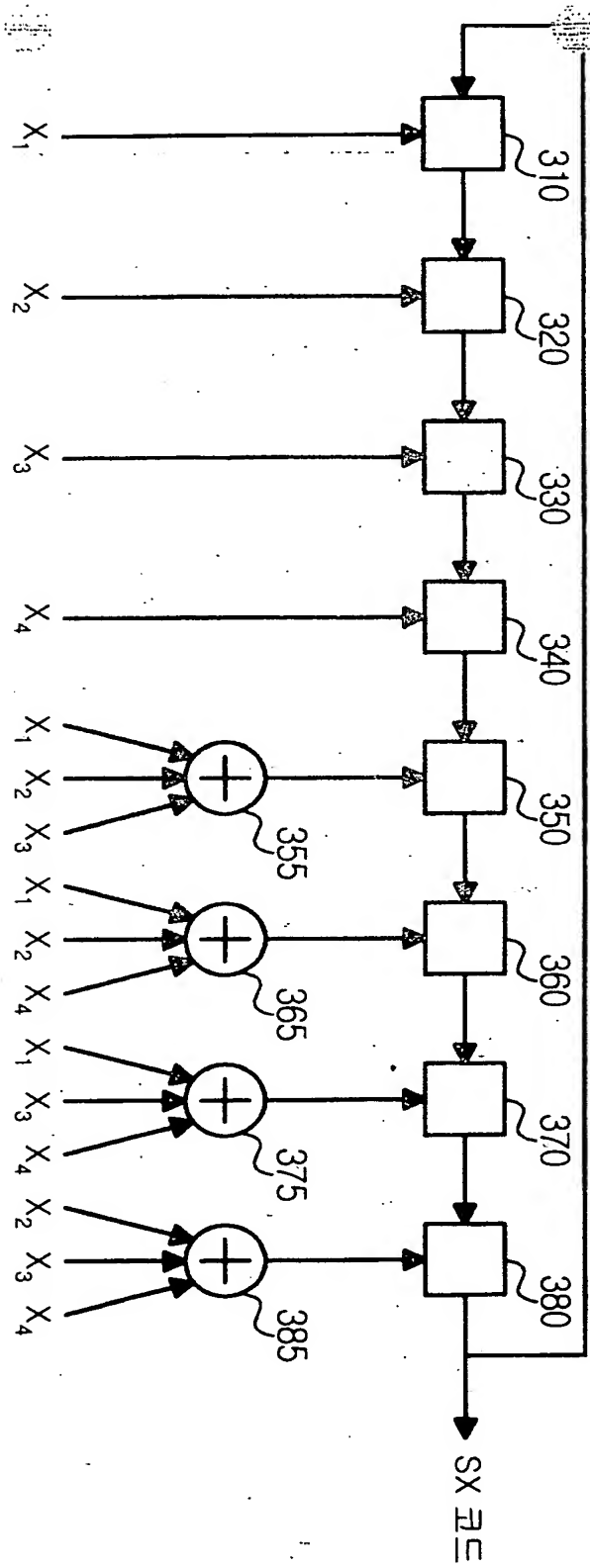
도면

도면 1





도면 3





입력	입력코드				출력								
	$X_1$	$X_2$	$X_3$	$X_4$		$F_1(X)$	$F_2(X)$	$F_3(X)$	$F_4(X)$	$F_5(X)$	$F_6(X)$	$F_7(X)$	$F_8(X)$
$X_0$	0	0	0	0	$SX_0$	0	0	0	0	0	0	0	0
$X_1$	0	0	0	1	$SX_1$	0	0	0	1	0	1	1	1
$X_2$	0	0	1	0	$SX_2$	0	0	1	0	1	0	1	1
$X_3$	0	0	1	1	$SX_3$	0	0	1	1	1	1	0	0
$X_4$	0	1	0	0	$SX_4$	0	1	0	0	1	1	0	1
$X_5$	0	1	0	1	$SX_5$	0	1	0	1	1	0	1	0
$X_6$	0	1	1	0	$SX_6$	0	1	1	0	0	1	1	0
$X_7$	0	1	1	1	$SX_7$	0	1	1	1	0	0	0	1
$X_8$	1	0	0	0	$SX_8$	1	0	0	0	1	1	1	0
$X_9$	1	0	0	1	$SX_9$	1	0	0	1	1	0	0	1
$X_{10}$	1	0	1	0	$SX_{10}$	1	0	1	0	0	1	0	1
$X_{11}$	1	0	1	1	$SX_{11}$	1	0	1	1	0	0	1	0
$X_{12}$	1	1	0	0	$SX_{12}$	1	1	0	0	0	0	1	1
$X_{13}$	1	1	0	1	$SX_{13}$	1	1	0	1	0	1	0	0
$X_{14}$	1	1	1	0	$SX_{14}$	1	1	1	0	1	0	0	0
$X_{15}$	1	1	1	1	$SX_{15}$	1	1	1	1	1	1	1	1

도면 5

	$F_1(X)$	$F_2(X)$
$SX_0$	0	0
$SX_1$	0	1

도면 6

	$F_1(X)$	$F_2(X)$	$F_3(X)$	$F_4(X)$
$SX_0$	0	0	0	0
$SX_1$	0	0	1	1
$SX_2$	0	1	0	1
$SX_3$	0	1	1	0

도면 7

	$F_1(X)$	$F_2(X)$	$F_3(X)$	$F_4(X)$	$F_5(X)$	$F_6(X)$	$F_7(X)$	$F_8(X)$
$SX_0$	0	0	0	0	0	0	0	0
$SX_1$	0	0	0	1	0	1	1	1
$SX_2$	0	0	1	0	1	0	1	1
$SX_3$	0	0	1	1	1	1	0	0
$SX_4$	0	1	0	0	1	1	0	1
$SX_5$	0	1	0	1	1	0	1	0
$SX_6$	0	1	1	0	0	1	1	0
$SX_7$	0	1	1	1	0	0	0	1

	$F_1(X)$	$F_2(X)$	$F_3(X)$	$F_4(X)$	$F_5(X)$	$F_6(X)$	$F_7(X)$	$F_8(X)$	$F_9(X)$	$F_{10}(X)$	$F_{11}(X)$	$F_{12}(X)$	$F_{13}(X)$	$F_{14}(X)$	$F_{15}(X)$	$F_{16}(X)$
$SX_0$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$SX_1$	0	0	0	0	1	0	0	1	0	1	1	0	1	1	1	1
$SX_2$	0	0	0	1	0	0	1	0	1	0	1	1	0	1	1	1
$SX_3$	0	0	0	1	1	0	1	1	1	1	0	1	1	0	0	0
$SX_4$	0	0	1	0	0	1	0	0	1	1	0	1	1	0	1	1
$SX_5$	0	0	1	0	1	1	0	1	1	0	1	1	0	1	0	0
$SX_6$	0	0	1	1	0	1	1	0	0	1	1	0	1	1	0	0
$SX_7$	0	0	1	1	1	1	1	1	0	0	0	0	0	0	1	1
$SX_8$	0	1	0	0	0	1	1	1	0	0	0	1	1	1	0	1
$SX_9$	0	1	0	0	1	1	1	0	0	1	1	1	0	0	1	0
$SX_{10}$	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
$SX_{11}$	0	1	0	1	1	1	0	0	1	1	0	0	0	1	0	1
$SX_{12}$	0	1	1	0	0	0	1	1	1	1	0	0	0	1	1	0
$SX_{13}$	0	1	1	0	1	0	1	0	1	0	1	0	1	0	0	1
$SX_{14}$	0	1	1	1	0	0	0	1	0	1	1	1	0	0	0	1
$SX_{15}$	0	1	1	1	1	0	0	0	0	0	0	1	1	1	1	0